CLAIMS

WHAT IS CLAIMED IS:

1. A method of searching for a string in a lexical cache, comprising the computer-implemented steps of:

generating a key based on the string;

identifying a lexical container from among a plurality of lexical containers based on a length of the key; and

searching the lexical container for an entry associated with the string.

2. The method of claim 1, wherein the step of generating a key based on the string includes the step of compressing the string to produce the key.

3. The method of claim 2, wherein the step of compressing the string to produce the key includes the step of performing an n-gram compression on the string.

4. The method of claim 1, wherein the step of generating a key based on the string includes the step of using the string as the key.

5. The method of claim 1, wherein the step of identifying a lexical container includes the steps of:

generating a prefix based on the key;

identifying the lexical container from among the plurality of the lexical containers based on the length of the key and the prefix.

6. The method of claim 1, wherein:

the step of identifying a lexical container based on a length of the key includes the step of identifying a hash table based on the length of the key, said hash table containing sequences of slots for holding entries associated with strings, each of said sequences of slots corresponding to a respective hash value; and

the step of searching the lexical container for an entry associated with said string includes the steps of:

computing a hash value based on the key; and

searching the hash table based on the hash value for a slot holding an entry associated with said string.

7. The method of claim 6, wherein the step of computing a hash value based on the key includes the step of computing the hash value based on the key and a prime number associated with the hash table.

8. The method of claim 7, wherein the step of searching the hash table based on the hash value includes the steps of:

indexing one or more fixed regions of the hash table, each of the fixed regions having the prime number of slots, based on the hash value to identify one or more respective slots; and

inspecting the one or more respective slots for a respective key value matching the key.

9. The method of claim 8, wherein the step of searching the hash table further includes the step of searching for the key in a linked list of slots stored in an expansion region of the hash table, if the key was not found in the one or more respective slots for the key.

10. The method of claim 6, further including the step of, if an entry for the string is not found at a first slot that corresponds to the hash value, but is found in a slot that belongs to a sequence of slots that correspond to keys that produce said hash value, then moving a relative position of the entry for the string within the sequence of slots toward the beginning of the sequence of slots.

11. The method of claim 6, further comprising the step of initializing a descriptor for the hash table, said descriptor storing a reference to the hash table and parameters for the hash table;

wherein the step of identifying a hash table includes the step of identifying a descriptor indicating the hash table and a prime number.

12. The method of claim 11, wherein the step of initializing a descriptor for the hash table includes the step of initializing a prime number for use in computing a hash value.

13. The method of claim 11, wherein the step of initializing a descriptor for the hash table includes the step of initializing a maximum number of slots for the hash table.

14. The method of claim 11, wherein the step of initializing a descriptor for the hash table includes the step of initializing a maximum length of the sequences of slots for the hash table.

15. A method of searching for a string in a lexical cache, comprising the computer-implemented steps of:

compressing the string to generate a key;

identifying a hash table from among a plurality of hash tables based on a length of the key, said hash table containing sequences of slots for holding respective key values, each of said sequences of slots corresponding to a respective hash value;

computing a hash value based on the key;

using said hash value to locate a beginning of the particular sequence of slots that correspond to said hash value;

searching the particular sequence of slots for a slot holding a key value matching the key; and

if a slot having a key value matching the key is found in the particular sequence of slots, but is not at the beginning of said particular sequence of slots, then moving a relative position of the key value within the particular sequence of slots toward the beginning of the particular sequence of slots.

16. A computer-readable medium bearing instructions for searching for a string in a lexical cache, said instructions arranged, when executed by one or more processors, to cause the one or more processors to perform the steps of:

generating a key based on the string;

identifying a lexical container from among a plurality of lexical containers based on a length of the key; and

searching the lexical container for an entry associated with the string.

17. The computer-readable medium of claim 16, wherein the step of generating a key based on the string includes the step of compressing the string to produce the key.

18. The computer-readable medium of claim 17, wherein the step of compressing the string to produce the key includes the step of performing an n-gram compression on the string.

19. The computer-readable medium of claim 16, wherein the step of generating a key based on the string includes the step of using the string as the key.

20. The computer-readable medium of claim 16, wherein the step of identifying a lexical container includes the steps of:

generating a prefix based on the key;

identifying the lexical container from among the plurality of the lexical containers based on the length of the key and the prefix.

21. The computer-readable medium of claim 16, wherein:

the step of identifying a lexical container based on a length of the key includes the step of identifying a hash table based on the length of the key, said hash table containing sequences of slots for holding entries associated with strings, each of said sequences of slots corresponding to a respective hash value; and

the step of searching the lexical container for an entry associated with said string includes the steps of:

computing a hash value based on the key; and

searching the hash table based on the hash value for a slot holding an entry associated with said string.

22. The computer-readable medium of claim 21, wherein the step of computing a hash value based on the key includes the step of computing the hash value based on the key and a prime number associated with the hash table.

23. The computer-readable medium of claim 22, wherein the step of searching the hash table based on the hash value includes the steps of:

indexing one or more fixed regions of the hash table, each of the fixed regions having the prime number of slots, based on the hash value to identify one or more respective slots; and

inspecting the one or more respective slots for a respective key value matching the key.

24. The computer-readable medium of claim 23, wherein the step of searching the hash table further includes the step of searching for the key in a linked list of slots stored in an expansion region of the hash table, if the key was not found in the one or more respective slots for the key.

25. The computer-readable medium of claim 21, wherein said instructions are further arranged to cause the one or more processors to perform the step of, if an entry for the string is not found at a first slot that corresponds to the hash value, but is found in a slot that belongs to a sequence of slots that correspond to keys that produce said hash value, then moving a relative position of the entry for the string within the sequence of slots toward the beginning of the sequence of slots.

26. The computer-readable medium of claim 21, wherein said instructions are further arranged to cause the one or more processors to perform the step of initializing a descriptor for the hash table, said descriptor storing a reference to the hash table and parameters for the hash table;

wherein the step of identifying a hash table includes the step of identifying a descriptor indicating the hash table and a prime number.

27. The computer-readable medium of claim 26, wherein the step of initializing a descriptor for the hash table includes the step of initializing a prime number for use in computing a hash value.

28. The computer-readable medium of claim 26, wherein the step of initializing a descriptor for the hash table includes the step of initializing a maximum number of slots for the hash table.

29. The computer-readable medium of claim 26, wherein the step of initializing a descriptor for the hash table includes the step of initializing a maximum length of the sequences of slots for the hash table.

30. A computer-readable medium bearing instructions for searching for a string in a lexical cache, said instructions arranged, when executed by one or more processors, to cause the one or more processors to perform the steps of:

compressing the string to generate a key;

identifying a hash table from among a plurality of hash tables based on a length of the key, said hash table containing sequences of slots for holding respective key values, each of said sequences of slots corresponding to a respective hash value;

computing a hash value based on the key;

using said hash value to locate a beginning of the particular sequence of slots that correspond to said hash value;

searching the particular sequence of slots for a slot holding a key value matching the key; and

if a slot having a key value matching the key is found in the particular sequence of slots, but is not at the beginning of said particular sequence of slots, then moving

a relative position of the key value within the particular sequence of slots toward the beginning of the particular sequence of slots.

Add 05

add 62

Add 07